BENJAMIN S. PETRILLO

petrillo.b@northeastern.edu | GitHub | LinkedIn | benpetrillo.dev | +1 339-788-6922 | Boston, MA 02115
Availability: July 2024 – December 2024

Motivated and proactive Computer Science student with a strong dedication to lifelong learning and enhancing technical skills. Thrives in both solo and collaborative settings; enthusiastic and self-driven. Delivers professional results in diverse environments.

EDUCATION

Northeastern University | Khoury College of Computer Sciences • Boston, MA — Sept 2022 – Present
Candidate for a Bachelor of Science in Computer Science | Concentration: Software — Expected May 2026

Honors: Dean's List

Relevant Coursework: Algorithms & Data Structures, Object-Oriented Design, Networks & Distributed Systems, Logic & Computation, Computer Systems, Programming in C++, Mathematics of Data Models

TECHNICAL KNOWLEDGE

Languages: Java, Python, TypeScript, C++, C, HTML, CSS, PHP, Racket, Common Lisp, SQL
Frameworks: Node.js, React, Express.js, JQuery, Flask, Numpy, PyTorch, Pandas, Cheerio, EJS, Tailwind
Technologies: MySQL, MongoDB, PostgreSQL, SQLite | **Cloud:** AWS EC2, Google Cloud, Docker, Nginx
Other Skills: Git, Linux, MacOS, Windows 10/11, GPG, Postman, Wireshark, Jetbrains, VSCode

WORK EXPERIENCE

Khoury College of Computer Sciences • Boston, MA
Teaching Assistant, Object-Oriented Design — May 2024 – July 2024
Teaching Assistant, Fundamentals of Computer Science II — Dec 2023 – May 2024

- Led and facilitated numerous weekly lab sections with over 60 students each, emphasizing course concepts.
- Graded exams, lab activities, and 30+ assignments per week, collaborating with professors and other staff members on grading rubrics.
- Contributed to curriculum development and lab lesson planning, preparing students for assignments and exams.
- Held weekly office hours, mentoring students one-on-one for comprehension of course concepts, assignment help, and data structures.
- Actively answered student questions on the class forum Piazza, prompting critical thinking and problem solving amongst students.

PROJECTS

Developer Portfolio | React, TypeScript, HTML, CSS, Vite, Google Cloud | GitHub Repository | View Live — Jun 2024 – Present

- Developed a dynamic and responsive portfolio website using React and TypeScript, achieving a 30% faster build performance with Vite. Implemented dynamic content management using JSON, enabling flexible content updates.
- Created a comment section with full authentication using Google Cloud OAuth 2.0 for secure user authentication, a RESTful API, and Sqlite3 for data storage, using AWS for deployment and Nginx for reverse proxies.
- Managed sessions for up to 1,000 users and enhanced security using HTTP-only cookies for storing authentication tokens, protecting against XSS attacks and leveraging cloud infrastructure for numerous concurrent users.

Elixir Music Bot & API | Java, TypeScript, Python, MongoDB | GitHub Repository — May 2021 – Present

- Engineered a sharded Discord bot application serving over 350,000 users, integrating the Spotify Tracks API and YouTube for audio streaming to Discord using Ffmpeg with queue pagination from application commands.
- Implemented a REST API providing real-time bot state by guild and using POST requests to modify the bot state externally. Used MongoDB for secure and encrypted user and custom playlist storage.
- Created an API wrapper in Python for third-party integrations and managed ISRC metadata for track identification.

Ponjo.Pastes | TypeScript, HTML, CSS, Express.js, MongoDB, EJS | GitHub Repository | View Live — Apr 2023 – Present

- Spearheaded the development of a full-stack code snippet sharing platform using MongoDB for code storage. Integrated a comprehensive search feature, with a public API and professional documentation using styled Vuepress.
- Designed a responsive web UI using HTML, CSS, and Bootstrap, integrating EJS to render snippets to the frontend.

Fakebook Web Crawler | Python | GitHub Repository — March 2024

- Devised a web crawler to traverse and find hidden flags from a large-scale mock social networking site Fakebook, using a queue-based system for managing visited and unvisited links, ensuring efficient navigation by avoiding duplicate visits.
- Optimized crawler performance by incorporating multithreading for concurrent HTTP requests, significantly reducing flag retrieval time from 30 minutes to 5 minutes. Made an HTML parser to accurately extract secret flags from user profiles.
- Developed robust error-handling mechanisms to handle HTTP status codes and maintain the crawler's reliability, including automatic retries for 503 status codes and 302 redirects.